

PATENT ABSTRACTS OF JAPAN

①

(11)Publication number : 2000-285090

(43)Date of publication of application : 13.10.2000

(51)Int.Cl.

G06F 15/177

G06F 13/16

G06F 13/36

(21)Application number : 11-092195

(71)Applicant : CANON APTEX INC

(22)Date of filing : 31.03.1999

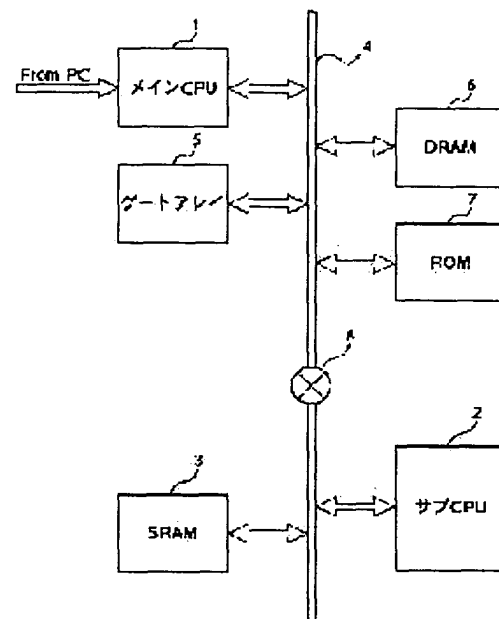
(72)Inventor : NAKAHARA YASUNORI
TAJIMA YUTAKA
OKUWAKI HIROTAKA
MIYAHARA FUMIO

(54) DATA PROCESSING METHOD AND SYSTEM

(57)Abstract:

PROBLEM TO BE SOLVED: To inexpensively provide data processing method and system of high speed and high performance by simple constitution without reducing performance and using an expensive dual port RAM.

SOLUTION: A main CPU 1 activates a bus right request signal in order to request the right of a system bus 4 to a sub-CPU 2, which activates a bus right request acknowledge signal in order to inform the main CPU 1 that the right of the bus 4 is released to an external device at the end of processing. A bus switching circuit 8 is provided which alternatively switches the disconnection and connection of the bus 4 between the main CPU 1 and the sub-CPU 2 by using the states of the bus right request signal and the bus right request acknowledge signal.



LEGAL STATUS

[Date of request for examination]

[Date of sending the examiner's decision of rejection]

[Kind of final disposal of application other than the examiner's decision of rejection or application converted registration]

[Date of final disposal for application]

[Patent number]

[Date of registration]

[Number of appeal against examiner's decision of rejection]

[Date of requesting appeal against examiner's decision of rejection]

[Date of extinction of right]

[Scope of Claims for Patent]

[Claim 1] A data processing method for processing data in a data processing system in which main and sub two CPUs (Central Processing Unit) commonly use one memory, the method comprising:

5 a first signal state switching step of switching a state of a bus right request signal with which the main CPU requests a right of a sub-system bus;

 a second signal state switching step of switching a state of a bus right request acknowledge signal which informs the main CPU that the right of the bus is released to an external device at the end of a process by the sub CPU;

10 and

 a bus switching step of alternatively switching disconnection and connection between a main system bus and the sub-system bus based on the states of the bus right request signal and the bus right request acknowledge signal.

15 [Claim 2] The data processing method according to claim 1, wherein, at the bus switching step, only when both the bus right request signal and the bus right request acknowledge signal are effective, the main system bus is connected with the sub-system bus.

20 [Claim 3] The data processing method according to claim 1, wherein, at the bus switching step, only when all the bus right request signal, the bus right request acknowledge signal, and the a CS signal at the main CPU are effective, the main system bus is connected with the sub-system bus.

25 [Claim 4] The data processing method according to claim 1, 2 or 3, wherein the bus switching step is executed by a three-state bidirectional buffer of TTL (Transistor-Transistor Logic).

[Claim 5] The data processing method according to claim 1, 2 or 3, wherein the bus switching step is executed by a three-state buffer of TTL (Transistor-Transistor Logic).

[Claim 6] The data processing method according to claim 4 or 5, wherein
5 the TTL is a CMOS (Complementary Metal-Oxide-Semiconductor) logic.

[Claim 7] A data processing system in which main and sub two CPUs (Central Processing Unit) commonly use one memory, the system comprising:

 a first signal state switching unit which switches a state of a bus right request signal with which the main CPU requests a right of a sub-system bus;

10 a second signal state switching unit which switches a state of a bus right request acknowledge signal which informs the main CPU that the right of the bus is released to an external device at the end of a process by the sub CPU; and

 a bus switching unit which alternatively switches disconnection and
15 connection between a main system bus and the sub-system bus based on the states of the bus right request signal and the bus right request acknowledge signal.

[Claim 8] The data processing system according to claim 7, wherein, only when both the bus right request signal and the bus right request
20 acknowledge signal are effective, the bus switching unit connects the main system bus with the sub-system bus.

[Claim 9] The data processing system according to claim 7, wherein, when all the bus right request signal, the bus right request acknowledge signal, and a CS signal at the main CPU are effective, the bus switching unit connects
25 the main system bus with the sub-system bus.

[Claim 10] The data processing system according to claim 7, 8 or 9, wherein the bus switching unit is composed of a three-state bidirectional buffer of TTL (Transistor-Transistor Logic).

[Claim 11] The data processing system according to claim 7, 8 or 9,
5 wherein the bus switching unit is composed of a three-state buffer of TTL (Transistor-Transistor Logic).

[Claim 12] The data processing system according to claim 10 or 11, wherein the TTL is a CMOS (Complementary Metal-Oxide-Semiconductor) logic.

10

[0002]

[Prior Art] When a new data processing function is added to a CPU in a currently existing system as an example of the first prior art, the new
15 processing function weighs on the current CPU, thereby greatly deteriorating the performance. In this case, the current CPU is replaced by a high performance CPU, so that a system to which the new processing function is added is structured without deteriorating the performance.

[0003] When a high speed and high performance system is originally
20 structured as an example of the second prior art, a plurality of CPUs are mounted and the CPUs can commonly use a memory, thereby heightening the process. A dual-port RAM (Random Access Memory) is connected between the memory which is commonly used for the CPUs and system buses of the CPUs, and writing and reading are carried out from the CPUs to the dual-port
25 RAMs, so that the high speed and high performance system is structured

simply.

[0004]

[Problems to be Solved by the Invention] In the example of the first prior art, however, the system can be structured simply, but all programs of software should be changed according to new CPUs, the schedule is extended greatly, thereby greatly delaying introduction of products in the market.

[0005] In the example of the second prior art, since a dual-port memory having large memory capacity does not exist, a plurality of dual-port memories should be used in order to structure a large-capacity system, thereby greatly increasing the cost.

[0006] The present invention has been achieved with a view of the problems of the prior arts, and its object is to provide inexpensive high speed and high performance data processing method and system with a simple structure without deteriorating a performance and using an expensive dual-port RAM.

[0020] Fig. 1 is a block diagram that shows a structure of a data processing system (Dual-CPU system) according to one embodiment of the present invention. In the drawing, 1 denotes a main CPU (Central Processing Unit), 2 denotes a sub CPU, 3 denotes an SRAM (Static Random Access Memory), 4 denotes a system bus (access signals to external devices such as address bus, data bus, read/write device, chip select device, data strobe device), 5 denotes a gate array, 6 denotes a DRAM (Dynamic Random Access Memory), 7 denotes a ROM (Read-Only Memory), and 8 denotes a bus switching circuit (bus switching unit).

[0021] The devices such as main CPU 1, the gate array 5, the DRAM 6 and the ROM 7 are connected with each other directly by the system bus 4, and connected also with the sub CPU 2 and the RAM 3 via the bus switching circuit 8.

- 5 [0022] The data processing system having such a structure allows the sub CPU 2 to execute a process which cannot be executed by the main CPU 1 or a troublesome process, and during this, the main CPU 1 can execute another process.

[0023] A flow of data expansion in the data processing system having the
10 structure shown in Fig. 1 is explained with reference to Figs. 2 to 7.

[0024] Fig. 2 is a flowchart of an operation for writing a command sent from a PC (personal Computer) into the SRAM 3 using the main CPU 1 or the gate array 5 in order to allow the sub CPU 2 to execute a process because an expansion process of the command takes a lot of time on the main CPU 1.

- 15 Access to the SRAM 3 may be executed directly by the main CPU 1, or when a terminal or the like is not sufficient, the control of the SRAM 3 may be accessed by an ASIC (Application Specific Integrated Circuit) such as the gate array 5.

[0025] After the command is written into the SRAM 3, both a bus right request signal (BREQ* signal), which is for requesting a right of the system bus
20 between the main CPU 1 and the sub CPU 2 and a bus right request acknowledge signal (BACK* signal), which is for informing the main CPU 1 that the right of the system bus is released to the external device at the end of a process by the sub CPU 2 are set into a nonactive state so that the system bus 4 is disconnected. Subsequently, the right of the system bus 4 is released to
25 the sub CPU 2.

[0026] The system bus 4 is released when the BACK* signal or the BREQ* signal is nonactive.

[0027] At the same time when the bus right request signal (BREQ* signal) is nonactive, the bus switching circuit 8 disconnects the system bus 4 from the
5 main and sub CPUs. This operation is shown in Fig. 3.

[0028] After the system bus 4 is disconnected, the sub CPU 2 executes the expansion process of the command written into the SRAM 3. This operation is shown in Fig. 4.

[0029] During this process, the main CPU 1 can execute another process
10 because it is disconnected from the system bus 4.

[0030] When the expansion process of the command in the SRAM 3 by the sub CPU 2 is ended, the sub CPU 2 transmits a command expansion process end signal that shows the command expansion process is ended to the main CPU 1. This operation is shown in Fig. 5.

15 [0031] When the main CPU 1 receives the command expansion process end signal, the main CPU 1 is again connected with the system bus 4 by the bus switching circuit 8 so as to set the bus right request signal (BREQ* signal) into an active state, and the sub CPU 2 sets the bus right request acknowledge signal (BACK* signal) to active. This operation is shown in Fig. 6.

20 [0032] It is not necessary to connect the system bus 4 with the main CPU 1 immediately after the command expansion process end signal is input into the main CPU 1.

[0033] When both the bus right request signal (BREQ* signal) and the bus right request acknowledge signal (BACK* signal) are active, the bus switching
25 circuit 8 connects the system bus 4 with the devices. The main CPU 1 or the

gate array 5 transmits the data in the SRAM 3 to the DRAM 6, and thus a series of the operation is completed. This operation is shown in Fig. 7.

[0034] Fig. 8 is a block diagram that shows a circuit configuration of the data processing system shown in Fig. 1, and in Fig. 8, the same parts as those in

5 Fig. 1 are designated by the same reference numbers.

[0035] In Fig. 8, 9a and 9b are data buses (system buses), 10a and 10b are address buses (system buses), 11a and 11b are R (read)/w (write) signals

(system buses), 12a and 12b are CS* signals (chip select, system bus), 13a and 13b are UDS* signals (data strobe, system bus), 14a and 14b are LDS*

10 signals (data strobe, system bus), 15 is an LS04 (inverter of TTL), 16 is an LS32 (OR circuit of TTL), 17 is a port (end) signal, 18 is the bus right request signal (BREQ* signal), 19 is the bus right request acknowledge signal (BREQ* signal), 20 is an LS32 (OR circuit of TTL), 21 is an LS32 (OR circuit of TTL), 22 is an LS245 (three-state bidirectional buffer of TTL), and 23 is an LS244
15 (three-state buffer of TTL).

[0036] In Fig. 8, the main CPU 1 writes the command which is received from an external interface into the SRAM 3, and the system bus 4 is disconnected

from the main CPU 1 and the sub CPU 2. The sub CPU 2 executes the expansion process of the command written into the SRAM 3, and when the

20 expansion process is ended, the sub CPU 2 transmits the command expansion process end signal 17 to the main CPU 1 so that the system bus 4 is again connected with them. The main CPU 1 reads data expanded in the SRAM 3.

[0037] The SRAM 3 used in this circuit has 4 Mbits (256 K × 16 bit), the data bus has 16 bits, and the address bus has 18 bits. The bus right request signal

25 (BREQ* signal) 18 and the bus right request acknowledge signal (BACK*

signal) 19 are connected with a BREQ* terminal and a BACK* terminal of the sub CPU 2, respectively, and the main CPU 1 is connected with an I/O (Input/output) port. Both the bus right request signal (BREQ* signal) 18 and the bus right request acknowledge signal (BACK* signal) 19 are pulled up so as to be set to nonactive in a high impedance state such as on resting.

5 [0038] When the bus right request signal (BREQ* signal) 18 output from the port of the main CPU 1 becomes active and is input into the sub CPU 2, the sub CPU 2 sets all outputs from the system buses (9a, 10b, 11b, 12b, 13b, and 14b) to high impedance and sets the bus right request acknowledge signal

10 (BACK* signal) 19 to active so as to output it to the main CPU 1. In such a manner, the system bus is connected.

[0039] The bus switching circuit which switches the system bus between release and connection includes "LS244: (three-state buffer of TTL)" 23 and "LS245: (three-state bidirectional buffer of TTL)" 22.

15 [0040] A terminal and B terminal as input terminals of "LS244" 23 are connected with the system bus on the main CPU 1, and YA terminal and YB terminal as output terminals are connected with the system bus on the sub CPU 2. Levels of ENABLE A terminal and ENABLE B terminal as bus switching signals are determined by an output signal from "LS32: (OR circuit of

20 TTL)" 20. The "LS32" 20 is an OR output of the bus right request signal (BREQ* signal) 18 and the bus right request acknowledge signal (BACK* signal) 19. When both the bus right request signal (BREQ* signal) 18 and the bus right request acknowledge signal (BACK* signal) 19 are active, namely, the main CPU 1 writes the command into the SRAM 3, the bus right request

25 signal (BREQ* signal) 18 is set to active (low level). Only when the sub CPU

2 sets the bus right request acknowledge signal (BACK* signal) 19 to active (low level), the output of "LS32" 20 becomes active (low level).

[0041] Only when the output of the "LS32" 20 is active, the system bus is connected with the main CPU 1 and the sub CPU 2, and the main CPU 1 can
5 write the command into the SRAM 3.

[0042] The switching by the bus switching circuit uses OR of the bus right request signal (BREQ* signal) 18 and the bus right request acknowledge signal (BACK* signal) 19 in order to obtain a safe circuit configuration where the system buss on the main CPU 1 and on the sub CPU 2 do not collide against
10 each other.

[0043] After the bus right request signal (BREQ* signal) 18 is active, and before the bus right request acknowledge signal (BACK* signal) 19 is active, the system bus of the sub CPU 2 has high impedance. For this reason, even if the sub CPU 2 outputs the bus right request acknowledge signal (BACK *
15 signal) 19 in the active state and a signal is sent from the main CPU 1 to the sub CPU 2, the system buses do not collide against each other. At the time when the bus right request signal (BREQ* signal) 18 becomes nonactive, namely, at the stage before the bus right request acknowledge signal (BACK* signal) 19 becomes nonactive, the system bus is disconnected from the main
20 CPU 1 and the sub CPU 2, thereby obtaining a circuit where the signals do not collide.

[0044] Fig. 9 is a timing chart that shows switching timing of the bus switching circuit.

[0045] The data buses 9a and 9b in the circuit are bidirectional signals, and
25 are switched by using the "LS245" 22. The A terminal of the "LS245" 22 is

connected with the system bus on the main CPU 1, and the B terminal is connected with the system bus on the sub CPU 2. A level of a G terminal as the bus switching signal is determined by an output signal of the "LS32" 21. The "LS32" 21 is an OR output of the bus right request signal (BREQ* signal) 18 and the bus right request acknowledge signal (BACK* signal) 19, and an OR output of the CS* signal 12a. When all the bus right request signal (BREQ* signal) 18, the bus right request acknowledge signal (BACK* signal) 19, and the CS* signal 12a are active, namely, when the main CPU 1 writes or reads the command into or from the SRAM 3 and only the CS* signal 11a output from the main CPU 1 is active, the output of the "LS32" 21 becomes active (low level).

[0046] Only when the output of the "LS32" 21 is active, the data bus 9a and the data bus 9b as the bidirectional buses are connected, so that the main CPU 1 can read/write the command from/into the SRAM 3.

15 [0047] In this embodiment, the switching signal of the system bus of the "LS245" 22 uses the "LS32" 21 for safety of the bus switching circuit, but in the configuration where no collision of the system buses occurs, the "LS32" 20 may switches the system bus between connection and disconnection.

[0048] Similarly, the "LS244" 23 may be switched by using the "LS32" 21.

20 [0049] A level of an I/O switching terminal DIR of the "LS245" 22 is determined by an output signal of the "LS32" 16. The "LS32" 16 outputs OR of a signal obtained in such a manner a R (read)/W (write) signal 11a as the output signal from the main CPU 1 is input into the "LS04: (inverter of TTL)" 15 and is inverted so as to be active when the R/W signal 11a is read, and the CS* signal 12a. When both the signals are active (low level), the switching

25

terminal DIR becomes low level, and the signal at the sub CPU 2 flows to the main CPU 1. That is to say, only when the main CPU 1 reads the data in the SRAM 3, the switching terminal DIR becomes low level.

[0050] Needless to say, the TTL logic used in the circuit explained in Fig. 8
5 may be a CMOS (Complementary Metal-Oxide-Semiconductor) logic or an equivalent logic.

[0051] The present invention is not limited only to products in which the bus right request signal and the bus right request acknowledge signal which are called as the BREQ* signal and the BACK* signal.

10 [0052] According to the circuit configuration, the data processing system according to the embodiment shown in Fig. 1 can be structured.

[0053]

[Effects due to the Invention] As detailed above, according to the present invention, the high speed and high performance data processing method and
15 system with a simple structure can be provided at low price without deteriorating the performance and using an expensive dual-port RAM.

[Brief Description of the Drawings]

[FIG. 1] Fig 1 is a block diagram that shows a structure of a data processing system according to one embodiment of the present invention.

20 [FIG. 2] Fig. 2 is a flowchart of data expansion in the data processing system according to one embodiment of the present invention.

[FIG. 3] Fig. 3 is a flowchart of the data expansion in the data processing system according to one embodiment of the present invention.

[FIG. 4] Fig. 4 is a flowchart of the data expansion in the data processing
25 system according to one embodiment of the present invention.

[FIG. 5] Fig. 5 is a flowchart of the data expansion in the data processing system according to one embodiment of the present invention.

[FIG. 6] Fig. 6 is a flowchart of the data expansion in the data processing system according to one embodiment of the present invention.

5 [FIG. 7] Fig. 7 is a flowchart of the data expansion in the data processing system according to one embodiment of the present invention.

[FIG. 8] Fig. 8 is a block diagram that shows a circuit diagram of the data processing system according to one embodiment of the present invention.

[FIG. 9] Fig. 9 is a timing chart that shows switching timing of a bus switching
10 circuit in the data processing system according to one embodiment of the present invention.

[Description of Signs]

1: main CPU

2: sub CPU

15 3: SRAM

4: system bus

5: gate array

6: DRAM

7: ROM

20 8: bus switching circuit

9a: data bus (system bus)

9b: data bus (system bus)

10a: address bus (system bus)

10b: address bus (system bus)

25 11a: R (read)/W (write) signal (system bus)

- 11b: R (read)/W (write) signal (system bus)
- 12a: CS* signal (chip select, system bus)
- 12b: CS* signal (chip select, system bus)
- 13a: UDS* signal (data strobe, system bus)
- 5 13b: USD* signal (data strobe, system bus)
- 14a: LDS* signal (data strobe, system bus)
- 14b: LDS* signal (data strobe, system bus)
- 15: LS04 (inverter of TTL)
- 16: LS32 (OR circuit of TTL)
- 10 17: port (end) signal
- 18: bus right request signal (BREQ* signal)
- 19: bus right request acknowledge signal (BACK* signal)
- 20: LS32 (OR circuit of TTL)
- 21: LS32 (OR circuit of TTL)
- 15 22: LS245 (three-state bidirectional buffer)
- 23: LS244 (three-state buffer of TTL)

FIG. 1

1: MAIN CPU

5: GATE ARRAY

2: SUB CPU

5

FIG. 2

1: MAIN CPU

5: GATE ARRAY

2: SUB CPU

10

FIG. 3

1: MAIN CPU

5: GATE ARRAY

BUS RELEASE

15 2: SUB CPU

FIG. 4

1: MAIN CPU

5: GATE ARRAY

20 2: SUB CPU

FIG. 5

1: MAIN CPU

5: GATE ARRAY

25 ① EXPANSION END

2: SUB CPU

FIG. 6

1: MAIN CPU

5 5: GATE ARRAY

① BUS CONNECTED

2: SUB CPU

FIG. 7

10 1: MAIN CPU

5: GATE ARRAY

2: SUB CPU

FIG. 8

15 1: MAIN CPU

2: SUB CPU

① Port (END SIGNAL)

FIG. 9

20 ① INPUT/OUTPUT SIGNAL ON THE SUB CPU

② ON RESETTING

③ MAIN CPU PERFORMS NORMAL OPERATION AND IS TRANSMITTING
COMMAND TO SRAM

④ SUB CPU IS EXECUTING EXPANSION AND MAIN CPU PERFORMS
25 NORMAL OPERATION (EXCEPT FOR SRAM)

- ⑤ MAIN CPU IS TRANSMITTING DATA TO DRAM, MAIN CPU PERFORMS NORMAL OPERATIONS AND IS TRANSMITTING COMMAND TO SRAM
- ⑥ SUB CPU IS EXECUTING EXPANSION AND MAIN CPU PERFORMS NORMAL OPERATION (EXCEPT FOR SRAM)

5

- ⑦ END SIGNAL (out)

- ⑧ BUS DISCONNECTED

- ⑨ BUS CONNECTED

10⑩ BUS DISCONNECTED

- ⑪ BUS CONNECTED

- ⑫ BUS DISCONNECTED

- ⑬ CONDITION OF BUS DISCONNECTION

15⑭ BUS

- ⑮ CONNECTED

- ⑯ DISCONNECTED

- ⑰ DISCONNECTED

- ⑱ DISCONNECTED

(19) 日本国特許庁 (J P)

(12) 公 開 特 許 公 報 (A)

(11) 特許出願公開番号
特開2000-285090
(P2000-285090A)

(43) 公開日 平成12年10月13日 (2000. 10. 13)

(51) Int.Cl. ⁷	識別記号	F I	テ-マ-ト (参考)
G 0 6 F 15/177	6 8 0	G 0 6 F 15/177	6 8 0 C 5 B 0 4 5
13/16	5 1 0	13/16	5 1 0 D 5 B 0 6 0
13/36	3 1 0	13/36	3 1 0 C 5 B 0 6 1

審査請求 未請求 請求項の数12 O L (全 9 頁)

(21) 出願番号 特願平11-92195

(22) 出願日 平成11年3月31日 (1999. 3. 31)

(71) 出願人 000208743

キヤノンアプテックス株式会社

茨城県水海道市坂手町5540-11

(72) 発明者 仲原 康則

茨城県水海道市坂手町5540-11 キヤノン

アプテックス株式会社内

(72) 発明者 田島 裕

茨城県水海道市坂手町5540-11 キヤノン

アプテックス株式会社内

(74) 代理人 100081880

弁理士 渡部 敏彦

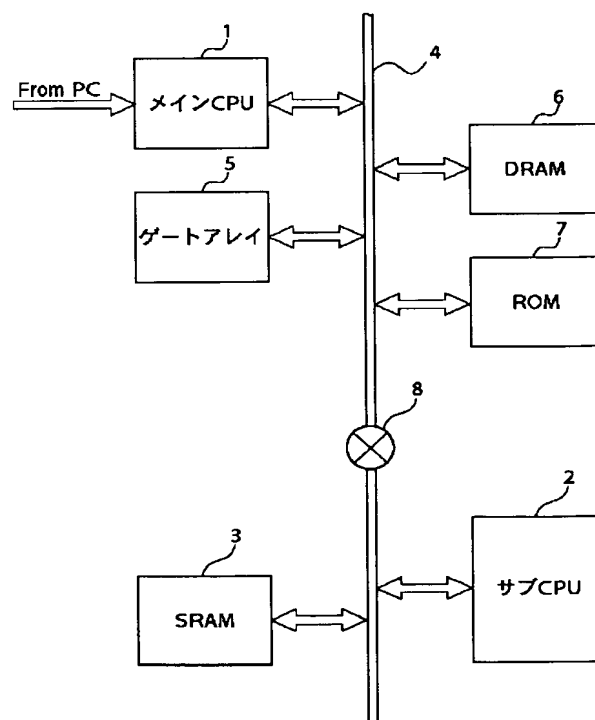
最終頁に続く

(54) 【発明の名称】 データ処理方法及びシステム

(57) 【要約】

【課題】 パフォーマンスを落とすことなく、また、高価なDual-Port-RAMを使用することなく、高速で高性能なデータ処理方法及びシステムを簡易な構成で安価に提供する。

【解決手段】 メインCPU 1がサブCPU 2にシステムバス4の権利を要求するためにバス権要求信号をアクティブにし、サブCPU 2が処理の終了した時点でシステムバス4の権利を外部デバイスに開放したことをメインCPU 1に通知するためにバス権要求アクノリッジ信号をアクティブとして、前記バス権要求信号及びバス権要求アクノリッジ信号の状態を用いてシステムバス4をメインCPU 1とサブCPU 2との間で切断状態と接続状態とに択一的に切り替えるバス切替回路8を有する。



【特許請求の範囲】

【請求項 1】 メイン側とサブ側の 2 つの CPU (Central Processing Unit: 中央処理装置) が 1 つのメモリを共有するデータ処理システムによりデータ処理するデータ処理方法であって、メイン側の CPU がサブ側のシステムバスの権利を要求するバス権要求信号の状態を切り替える第 1 の信号状態切替工程と、サブ側の CPU の処理が終了した時点で前記バスの権利を外部デバイスに開放したことを前記メイン側の CPU に通知するバス権要求アクノリッジ信号の状態を切り替える第 2 の信号状態切替工程と、前記バス権要求信号及び前記バス権要求アクノリッジ信号の状態に基づいてメイン側のシステムバスと前記サブ側のシステムバスとの間を切断状態と接続状態とに択一的に切り替えるバス切替工程とを有することを特徴とするデータ処理方法。

【請求項 2】 前記バス切替工程は、前記バス権要求信号と前記バス権要求アクノリッジ信号とが共に有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする請求項 1 記載のデータ処理方法。

【請求項 3】 前記バス切替工程は、前記バス権要求信号とバス権要求アクノリッジ信号及び前記メイン側の CPU の CS 信号の全てが有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする請求項 1 記載のデータ処理方法。

【請求項 4】 前記バス切替工程は、TTL (Transistor-Transistor Logic: トランジスタ・トランジスタ論理回路) の 3 ステートの双方向のバッファにより行うことを特徴とする請求項 1、2 または 3 記載のデータ処理方法。

【請求項 5】 前記バス切替工程は、TTL (Transistor-Transistor Logic) の 3 ステートバッファにより行うことを特徴とする請求項 1、2 または 3 記載のデータ処理方法。

【請求項 6】 前記 TTL は、CMOS (Complementary Metal-Oxide-Semiconductor: 相補型金属酸化膜半導体) ロジックであることを特徴とする請求項 4 または 5 記載のデータ処理方法。

【請求項 7】 メイン側とサブ側の 2 つの CPU (Central Processing Unit: 中央処理装置) が 1 つのメモリを共有するデータ処理システムであって、メイン側の CPU がサブ側のシステムバスの権利を要求するバス権要求信号の状態を切り替える第 1 の信号状態切替手段を備え、サブ側の CPU が、サブ側の処理が終了した時点で前記バスの権利を外部デバイスに開放したことを前記メイン側の CPU に通知するバス権要求アクノリッジ信号の状態を切り替える第 2 の信号

状態切替手段を備え、前記バス権要求信号及び前記バス権要求アクノリッジ信号の状態に基づいてメイン側のシステムバスと前記サブ側のシステムバスとの間を切断状態と接続状態とに択一的に切り替えるバス切替手段を有することを特徴とするデータ処理システム。

【請求項 8】 前記バス切替手段は、前記バス権要求信号と前記バス権要求アクノリッジ信号とが共に有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする請求項 7 記載のデータ処理システム。

【請求項 9】 前記バス切替手段は、前記バス権要求信号と前記バス権要求アクノリッジ信号及び前記メイン側の CPU の CS 信号の全てが有効時のみ、前記メイン側の前に前記サブ側のシステムバスを開放するように切り替えることを特徴とする請求項 7 記載のデータ処理システム。

【請求項 10】 前記バス切替手段は、TTL (Transistor-Transistor Logic: トランジスタ・トランジスタ論理回路) の 3 ステートの双方向のバッファにより構成したことを特徴とする請求項 7、8 または 9 記載のデータ処理システム。

【請求項 11】 前記バス切替手段は、TTL (Transistor-Transistor Logic) の 3 ステートバッファにより構成したことを特徴とする請求項 7、8 または 9 記載のデータ処理システム。

【請求項 12】 前記 TTL は、CMOS (Complementary Metal-Oxide-Semiconductor: 相補型金属酸化膜半導体) ロジックであることを特徴とする請求項 10 または 11 記載のデータ処理システム。

【発明の詳細な説明】

【0001】

【発明の属する技術分野】本発明は、メイン側とサブ側の 2 つの CPU (Central Processing Unit: 中央処理装置) が 1 つのメモリを共有するデータ処理システムにより処理するデータ処理方法及びシステムに関する。

【0002】

【従来技術】第 1 従来例として、現在存在するシステムで新たなデータ処理機能を CPU に追加する場合、その新たな処理機能が現状の CPU にとって重荷となり、パフォーマンスが大きく落ち込んでしまう場合がある。このような場合、現在の CPU より高性能な CPU に取り替えることにより、パフォーマンスを落とすことなく、新たな処理機能を追加したシステムを構築していた。

【0003】また、第 2 従来例として、当初より高速、高性能なシステムを構築する場合には、複数の CPU を搭載し且つメモリを複数の CPU で共有できる構成にして処理の高速化を図っていた。その際の複数の CPU で

共有するメモリは、それぞれのCPUのシステムバス間にDual-Port-RAM(Random Access Memory:デュアルポート読取り書き込み記憶装置)を接続し、それぞれのCPUからDual-Port-RAMへ書き込み、読み出しを行うことによって、高速で高性能なシステムを簡易構成で構築していた。

【0004】

【発明が解決しようとする課題】しかしながら、上述した第1従来例にあつては、システム構成を簡易にすることはできるが、ソフトウェアのプログラムを全て新たなCPUの仕様に変更しなければならず、日程の大幅な延長となるため、製品の市場投入が大幅に遅れてしまうという問題点があった。

【0005】また、上述した第2従来例にあつては、メモリ容量の大きなDual-Port-Memory(デュアルポートメモリ)自体が無く、大容量のシステムを構成するためには、複数のDual-Port-Memoryを用いなければならず、しかも多大なコストアップとなってしまうという問題点があった。

【0006】本発明は上述した従来の技術の有するこのような問題点に鑑みてなされたものであり、その目的とするところは、パフォーマンスを落とすことなく、また、高価なDual-Port-RAMを使用することなく、高速で高性能なデータ処理方法及びシステムを簡易な構成で安価に提供しようとするものである。

【0007】

【課題を解決するための手段】上記目的を達成するために請求項1記載のデータ処理方法は、メイン側とサブ側の2つのCPU(Central Processing Unit:中央処理装置)が1つのメモリを共有するデータ処理システムによりデータを処理するデータ処理方法であつて、メイン側のCPUがサブ側のシステムバスの権利を要求するバス権要求信号の状態を切り替える第1の信号状態切替工程と、サブ側のCPUの処理が終了した時点で前記バスの権利を外部デバイスに開放したことを前記メイン側のCPUに通知するバス権要求アクノリッジ信号の状態を切り替える第2の信号状態切替工程と、前記バス権要求信号及び前記バス権要求アクノリッジ信号の状態に基づいてメイン側のシステムバスと前記サブ側のシステムバスとの間を切断状態と接続状態とに択一的に切り替えるバス切替工程とを有することを特徴とする。

【0008】また、上記目的を達成するために請求項2記載のデータ処理方法は、請求項1記載のデータ処理方法において、前記バス切替工程は、前記バス権要求信号と前記バス権要求アクノリッジ信号とが共に有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする。

【0009】また、上記目的を達成するために請求項3

記載のデータ処理方法は、請求項1記載のデータ処理方法において、前記バス切替工程は、前記バス権要求信号と前記バス権要求アクノリッジ信号及び前記メイン側のCPUのCS信号の全てが有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする。

【0010】また、上記目的を達成するために請求項4記載のデータ処理方法は、請求項1、2または3記載のデータ処理方法において、前記バス切替工程は、TTL(Transistor-Transistor Logic:トランジスタ・トランジスタ論理回路)の3ステートの双方向のバッファにより行うことを特徴とする。

【0011】また、上記目的を達成するために請求項5記載のデータ処理方法は、請求項1、2または3記載のデータ処理方法において、前記バス切替工程は、TTL(Transistor-Transistor Logic)の3ステートバッファにより行うことを特徴とする。

【0012】また、上記目的を達成するために請求項6記載のデータ処理方法は、請求項4または5記載のデータ処理方法において、前記TTLは、CMOS(Complementary Metal-Oxide-Semiconductor:相補型金属酸化膜半導体)ロジックであることを特徴とする。

【0013】また、上記目的を達成するために請求項7記載のデータ処理システムは、メイン側とサブ側の2つのCPU(Central Processing Unit:中央処理装置)が1つのメモリを共有するデータ処理システムであつて、メイン側のCPUがサブ側のシステムバスの権利を要求するバス権要求信号の状態を切り替える第1の信号状態切替手段を備え、サブ側のCPUが、サブ側の処理が終了した時点で前記バスの権利を外部デバイスに開放したことを前記メイン側のCPUに通知するバス権要求アクノリッジ信号の状態を切り替える第2の信号状態切替手段を備え、前記バス権要求信号及び前記バス権要求アクノリッジ信号の状態に基づいてメイン側のシステムバスと前記サブ側のシステムバスとの間を切断状態と接続状態とに択一的に切り替えるバス切替手段を有することを特徴とする。

【0014】また、上記目的を達成するために請求項8記載のデータ処理システムは、請求項7記載のデータ処理システムにおいて、前記バス切替手段は、前記バス権要求信号と前記バス権要求アクノリッジ信号とが共に有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを接続するように切り替えることを特徴とする。

【0015】また、上記目的を達成するために請求項9記載のデータ処理システムは、請求項7記載のデータ処理システムにおいて、前記バス切替手段は、前記バス権

要求信号と前記バス権要求アクノリッジ信号及び前記メイン側のCPUのCS信号の全てが有効時のみ、前記メイン側のシステムバスに前記サブ側のシステムバスを開放するように切り替えることを特徴とする。

【0016】また、上記目的を達成するために請求項10記載のデータ処理システムは、請求項7、8または9記載のデータ処理システムにおいて、前記バス切替手段は、TTL (Transistor-Transistor Logic: トランジスタ・トランジスタ論理回路) の3ステートの双方向のバッファにより構成したことを特徴とする。

【0017】また、上記目的を達成するために請求項11記載のデータ処理システムは、請求項7、8または9記載のデータ処理システムにおいて、前記バス切替手段は、TTL (Transistor-Transistor Logic) の3ステートバッファにより構成したことを特徴とする。

【0018】また、上記目的を達成するために請求項12記載のデータ処理システムは、請求項10または11記載のデータ処理システムにおいて、前記TTLは、CMOS (Complementary Metal-Oxide-Semiconductor: 相補型金属酸化膜半導体) ロジックであることを特徴とする。

【0019】

【発明の実施の形態】以下、本発明の一実施の形態を図面に基づき説明する。

【0020】図1は、本発明の一実施の形態に係るデータ処理システム (Dual-CPU装置システム) の構成を示すブロック図であり、同図において、1はメインCPU (中央処理装置)、2はサブCPU (中央処理装置)、3はSRAM (Static Random Access Memory: スタティックランダムアクセスメモリ)、4はシステムバス (アドレスバス、データバス、リード/ライト、チップセレクト、データストロープ等の外部デバイスへのアクセス信号)、5はゲートアレイ、6はDRAM (Dynamic Random Access Memory: ダイナミックランダムアクセスメモリ)、7はROM (リードオンリーメモリ)、8はバス切替回路 (バス切替手段) である。

【0021】そして、メインCPU1、ゲートアレイ5、DRAM6及びROM7等のデバイスがシステムバス4により直接接続され且つバス切替回路8を介してサブCPU2及びSRAM3へも接続されている。

【0022】このような構成のデータ処理システムは、メインCPU1で対応できない、または手間がかかる処理をサブCPU2で行わせて、その間メインCPU1側では別の処理を実行することができるものである。

【0023】図1に示した構成のデータ処理システムにおけるデータ展開の流れを、図2～図7を用いて説明する。

【0024】図2は、PC (Personal Computer: パーソナルコンピュータ) から送られたコマンドがメインCPU1側では展開処理に時間がかかるので、サブCPU2側に処理を任せるため、メインCPU1、またはゲートアレイ5を用いてSRAM3に書き込む動作を示した図である。SRAM3へのアクセス方法は、メインCPU1が直接行っても良いし、端子等が足りない場合はSRAM3の制御をゲートアレイ5等のASIC (Application Specific Integrated Circuit: 特定用途向け集積回路) でアクセスする方法もある。

【0025】SRAM3にコマンドを書き終わった後は、メインCPU1とサブCPU2との間のシステムバスの権利を要求する信号であるバス権要求信号 (BREQ*信号) 及びサブCPU2側の処理が終了した時点で前記システムバスの権利を外部デバイスに開放したことをメインCPU1に通知する信号であるバス権要求アクノリッジ信号 (BACK*信号) を共に非アクティブとしてシステムバス4を切り離し、サブCPU2にシステムバス4の権利を開放する。

【0026】システムバス4の開放はBACK*信号またはBREQ*信号のどちらかが非アクティブの時である。

【0027】前記バス権要求信号 (BREQ*信号) が非アクティブになると同時に、バス切替回路8によってシステムバス4がメイン側とサブ側に切断される。この動作を示したのが図3である。

【0028】システムバス4が切断された後、サブCPU2はSRAM3にライト (書き込み) されたコマンドの展開処理を行う。この動作を示したのが図4である。

【0029】その間、メインCPU1側は、システムバス4が切断されているので、別の処理を実行することが可能である。

【0030】サブCPU2によるSRAM3のコマンドの展開処理が終了すると、メインCPU1にコマンド展開処理が終了したことを示すコマンド展開処理終了信号を送る。この動作を示したのが図5である。

【0031】メインCPU1がコマンド展開処理終了信号を受信すると、該メインCPU1は再びバス切替回路8によってシステムバス4を接続するため、前記バス権要求信号 (BREQ*信号) をアクティブにし、サブCPU2もバス権要求アクノリッジ信号 (BACK*信号) をアクティブにする。この動作を示したのが図6である。

【0032】但し、メインCPU1にコマンド展開処理終了信号が入力されてすぐにシステムバス4を接続する必要はない。

【0033】前記バス権要求信号 (BREQ*信号) 及びバス権要求アクノリッジ信号 (BACK*信号) が共にアクティブになると、バス切替回路8によってシステ

ムバス4が接続される。そして、メインCPU1、またはゲートアレイ5がSRAM3のデータをDRAM6に転送することによって一連の動作が終了する。この動作を示したのが図7である。

【0034】図8は、図1に示すデータ処理システムの回路構成を示すブロック図であり、同図において図1と同一部分には同一符号が付してある。

【0035】図8において、9a、9bはデータバス（システムバス）、10a、10bはアドレスバス（システムバス）、11a、11bはR（リード）／W（ライト）信号（システムバス）、12a、12bはCS*信号（チップセレクト、システムバス）、13a、13bはUDS*信号（データストローブ、システムバス）、14a、14bはLDS*信号（データストローブ、システムバス）、15はLS04（TTLのインバータ）、16はLS32（TTLのOR回路）、17はport（終了）信号、18はバス権要求信号（BREQ*信号）、19はバス権要求アクノリッジ信号（BACK*信号）、20はLS32（TTLのOR回路）、21はLS32（TTLのOR回路）、22はLS245（TTLの3ステート双方向バッファ）、23はLS244（TTLの3ステートバッファ）である。

【0036】図8において、まず、メインCPU1が外部インターフェースから受け取ったコマンドをSRAM3に書き込み、その後、メインCPU1とサブCPU2との間でシステムバス4を切断し、サブCPU2でSRAM3に書き込まれたコマンドの展開処理を行い、その展開処理が終了すると、サブCPU2がメインCPU1にコマンド展開処理終了信号17を送り、再びシステムバス4を接続させ、メインCPU1がSRAM3に展開されたデータをリード（読み出し）するという回路構成である。

【0037】本回路で使用しているSRAM3は、4Mbit（256K×16bit）であり、データバスは16bit、アドレスバスは18bitである。バス権要求信号（BREQ*信号）18及びバス権要求アクノリッジ信号（BACK*信号）19は、それぞれサブCPU2のBREQ*端子及びBACK*端子に接続し、メインCPU1側にはI/O（入出力）ポートを接続する。そして、バス権要求信号（BREQ*信号）18及びバス権要求アクノリッジ信号（BACK*信号）19の両信号共にプルアップをして、リセット中等のハイインピーダンス状態のときは非アクティブになるようにする。

【0038】メインCPU1のポートから出力されるバス権要求信号（BREQ*信号）18がアクティブになってサブCPU2に入力されると、該サブCPU2はシステムバス（9a、10b、11b、12b、13b、14b）の出力を全てハイインピーダンスにして、バス権要求アクノリッジ信号（BACK*信号）19をアク

ティブにして、メインCPU1に出力することによって、システムバスの接続が行われる。

【0039】システムバスの開放及び接続の切り替えを行うバス切替回路は、「LS244：（TTLの3ステートバッファ）」23、「LS245：（TTLの3ステート双方向バッファ）」22によって構成される。

【0040】「LS244」23の入力端子であるA端子及びB端子にはメインCPU1側のシステムバスを接続し、出力端子であるYA端子及びYB端子にはサブCPU2側のシステムバスを接続する。そして、バスの切り替え信号であるENABLEA端子及びENABLEB端子のレベルは、「LS32：（TTLのOR回路）」20の出力信号によって決定される。この「LS32」20は、バス権要求信号（BREQ*信号）18とバス権要求アクノリッジ信号（BACK*信号）19のORの出力であり、バス権要求信号（BREQ*信号）18とバス権要求アクノリッジ信号（BACK*信号）19が共にアクティブのとき、即ちメインCPU1がSRAM3へコマンドの書き込みを行う時に、バス権要求信号（BREQ*信号）18をアクティブ（ローレベル）にして、サブCPU2もバス権要求アクノリッジ信号（BACK*信号）19をアクティブ（ローレベル）にしたときのみ、「LS32」20の出力はアクティブ（ローレベル）になる。

【0041】そして、「LS32」20の出力がアクティブのときのみ、メインCPU1とサブCPU2との間のシステムバスが接続され、メインCPU1がSRAM3へコマンドの書き込みを行うことができる。

【0042】バス切替回路の切り替えにバス権要求信号（BREQ*信号）18とバス権要求アクノリッジ信号（BACK*信号）19のORを用いるのは、メインCPU1側とサブCPU2側のシステムバスが衝突しない安全な回路構成にするためである。

【0043】バス権要求信号（BREQ*信号）18がアクティブになり、その後、バス権要求アクノリッジ信号（BACK*信号）19がアクティブになる前に、サブCPU2のシステムバスは、ハイインピーダンスになっているので、バス権要求アクノリッジ信号（BACK*信号）19をアクティブで出力して、メインCPU1側から信号が流れてきても、システムバスの衝突はなく、また、バス権要求信号（BREQ*信号）18が非アクティブになった時点で、即ちバス権要求アクノリッジ信号（BACK*信号）19が非アクティブになる前の段階で、メインCPU1とサブCPU2の間でシステムバスを切断することによって、信号の衝突しない回路にすることができる。

【0044】図9は、バス切替回路の切り替えタイミングを示すタイミングチャートである。

【0045】また、本回路のデータバス9a、9bは双方向信号であり、「LS245」22を用いて切り替え

る。この「LS245」22のA端子にはメインCPU1側のシステムバスを接続し、B端子にはサブCPU2側のシステムバスを接続する。そして、バスの切り替え信号であるG端子のレベルは、「LS32」21の出力信号によって決定される。この「LS32」21は、バス権要求信号(BREQ*信号)18とバス権要求アクノリッジ信号(BACK*信号)19のORの出力と、CS*信号12aのORの出力であり、バス権要求信号(BREQ*信号)18及びバス権要求アクノリッジ信号(BACK*信号)19及びCS*信号12aが全てアクティブのとき、即ちメインCPU1がSRAM3へコマンドのライトまたはリード時で且つメインCPU1が出力するCS*信号11aがアクティブのときのみ、「LS32」21の出力はアクティブ(ローレベル)になる。

【0046】そして、「LS32」21の出力がアクティブのときのみ、双方向バスであるデータバス9aとデータバス9bとが接続され、メインCPU1がSRAM3へコマンドのリード/ライトを行うことができる。

【0047】なお、本実施の形態では、「LS245」22のシステムバスの切り替え信号は、バス切替回路の安全性のため「LS32」21を用いているが、システムバスの衝突の恐れがない構成であれば「LS32」20によってシステムバスの切断及び接続の切り替えを行っても良い。

【0048】また、同様に「LS244」23の切り替えも「LS32」21を用いて行っても良い。

【0049】そして、「LS245」22の入出力の切り替え端子DIRのレベルは、「LS32」16の出力信号によって決定される。この「LS32」16は、メインCPU1の出力信号のR(リード)/W(ライト)信号11aを「LS04:(TTLのインバータ)」15に入力してR/W信号11aがリードのときはアクティブになるように反転させた信号と、CS*信号12aのORを出力し、両信号共にアクティブ(ローレベル)のとき、切り替え端子DIRはローレベルになり、サブCPU2側の信号がメインCPU1側に流れる。即ちメインCPU1がSRAM3のデータをリードする場合においてのみ、切り替え端子DIRはローレベルとなる。

【0050】なお、図8で説明した回路で使用しているTTLロジックは、CMOS(Complementary Metal-Oxide-Semiconductor: 相補型金属酸化膜半導体)ロジックまたはこれと同等のロジックであれば良いことは言うまでもない。

【0051】また、本発明は、バス権要求信号及びバス権要求アクノリッジ信号を、BREQ*信号及びBACK*信号と呼ぶ製品のものに限定されるものではない。

【0052】以上詳述したような回路構成によって、図1に示す本実施の形態に係るデータ処理システムを構築することができる。

【0053】

【発明の効果】以上詳述したように本発明によれば、パフォーマンスを落とすことなく、また、高価なDual-Port-RAMを使用することなく、高速で高性能なデータ処理方法及びシステムを簡易な構成で安価に提供することができるという効果を奏する。

【図面の簡単な説明】

【図1】本発明の一実施の形態に係るデータ処理システムの構成を示すブロック図である。

10 【図2】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

【図3】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

【図4】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

【図5】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

【図6】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

20 【図7】本発明の一実施の形態に係るデータ処理システムにおけるデータ展開の流れを示す図である。

【図8】本発明の一実施の形態に係るデータ処理システムの回路構成を示すブロック図である。

【図9】本発明の一実施の形態に係るデータ処理システムにおけるバス切替回路の切り替えタイミングを示すタイミングチャートである。

【符号の説明】

- | | |
|---------|-------------------------------|
| 1 | メインCPU |
| 2 | サブCPU |
| 30 3 | SRAM |
| 4 | システムバス |
| 5 | ゲートアレイ |
| 6 | DRAM |
| 7 | ROM |
| 8 | バス切替回路 |
| 9 a | データバス (システムバス) |
| 9 b | データバス (システムバス) |
| 10 a | アドレスバス (システムバス) |
| 10 b | アドレスバス (システムバス) |
| 40 11 a | R (リード) / W (ライト) 信号 (システムバス) |
| 11 b | R (リード) / W (ライト) 信号 (システムバス) |
| 12 a | CS*信号 (チップセレクト、システムバス) |
| 12 b | CS*信号 (チップセレクト、システムバス) |
| 13 a | UDS*信号 (データストロープ、システムバス) |
| 13 b | UDS*信号 (データストロープ、システムバス) |
| 50 14 a | LDS*信号 (データストロープ、システムバス) |

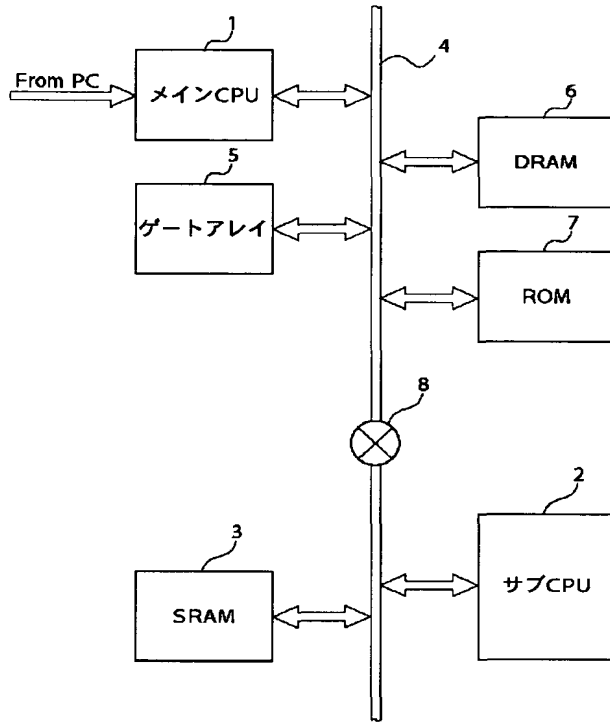
11

12

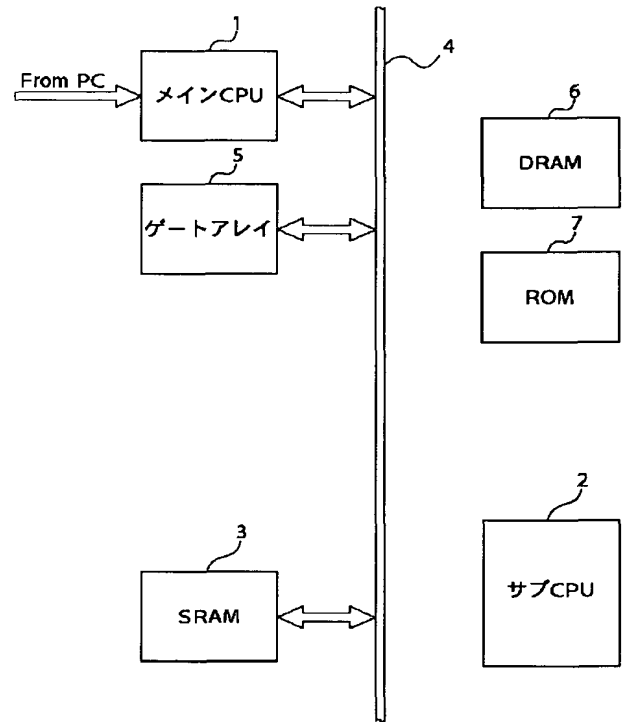
- ス)
 14 b LDS*信号(データストロブ、システムバス)
 15 LS04(TTLのインバータ)
 16 LS32(TTLのOR回路)
 17 port(終了)信号
 18 バス権要求信号(BREQ*信号)

- 19 バス権要求アクノリッジ信号(BACK*信号)
 20 LS32(TTLのOR回路)
 21 LS32(TTLのOR回路)
 22 LS245(TTLの3ステート双方向バッファ)
 23 LS244(TTLの3ステートバッファ)

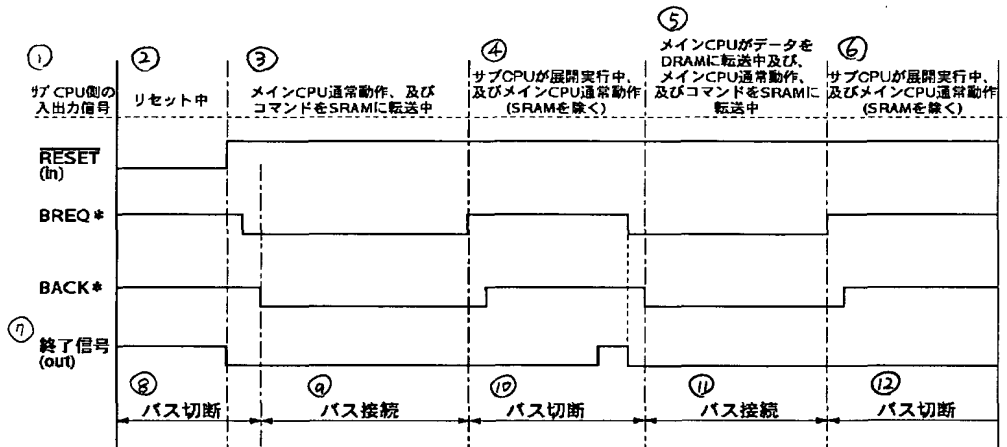
【図1】



【図2】



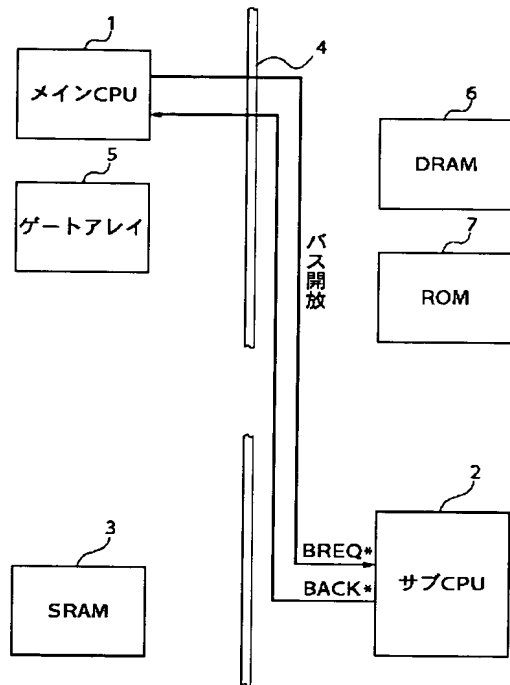
【図9】



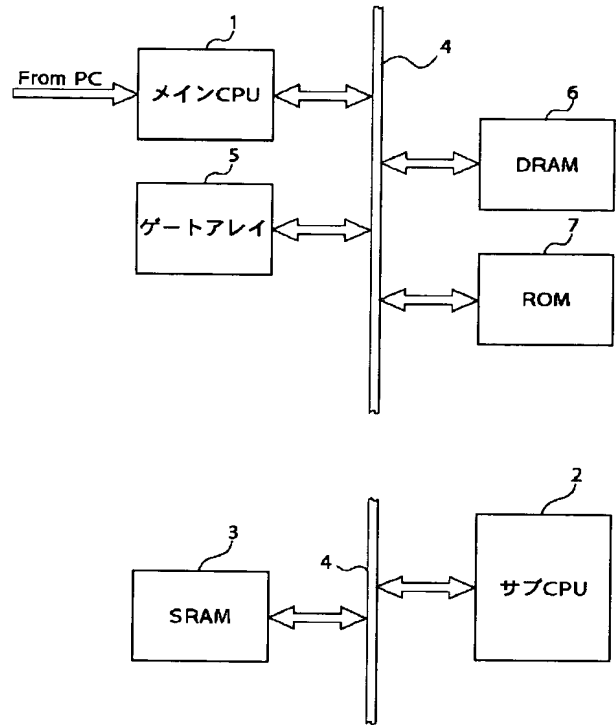
⑬ バス切断条件

BREQ*	BACK*	バス	⑭
'L'	'L'	接続	⑮
'L'	'H'	切断	⑯
'H'	'L'	切断	⑰
'H'	'H'	切断	⑱

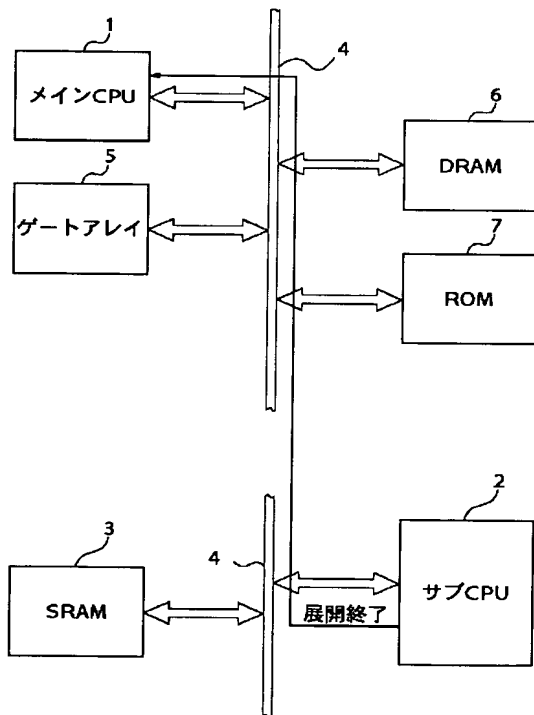
【図 3】



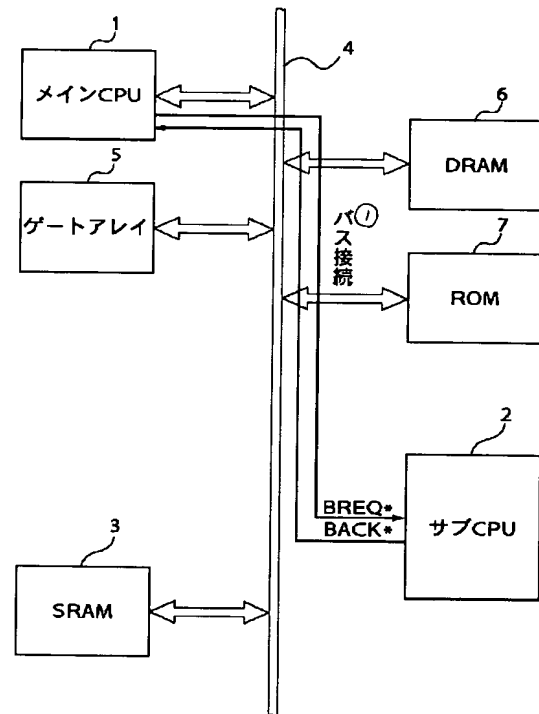
【図 4】



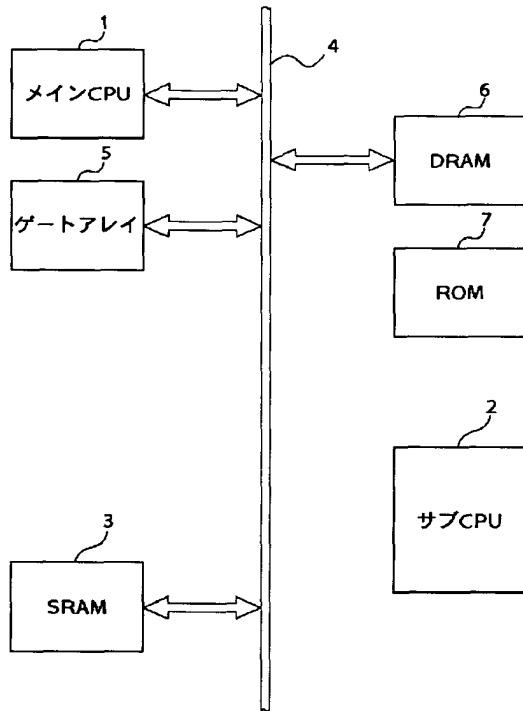
【図 5】



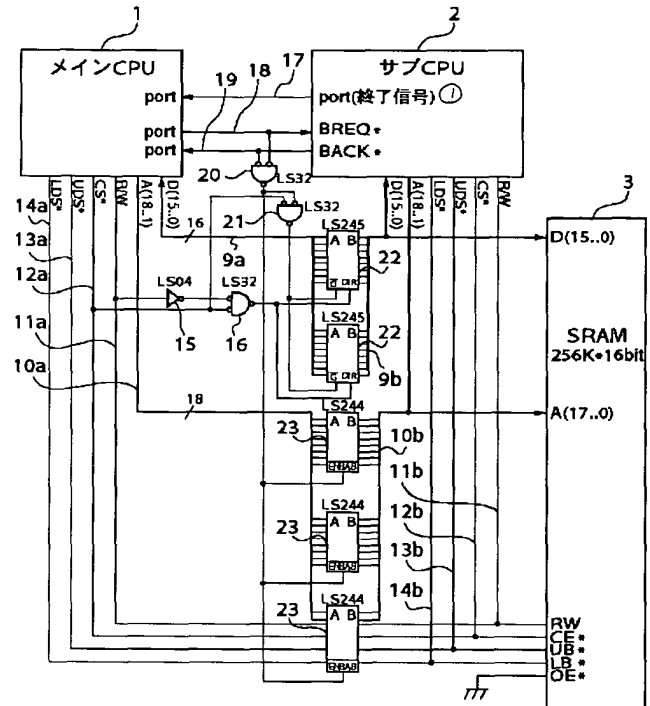
【図 6】



【図7】



【図8】



フロントページの続き

(72) 発明者 奥脇 裕貴
茨城県水海道市坂手町5540-11 キヤノン
アプテックス株式会社内

(72) 発明者 宮原 文雄
茨城県水海道市坂手町5540-11 キヤノン
アプテックス株式会社内

F ターム(参考) 5B045 BB14 BB36 EE07

5B060 KA02 KA04 MB01

5B061 FF01 FF23 GG13 RR02 RR03